A Method for Semantic Web Service Selection Based on QoS Ontology

Guangjun Guo
Department of Computer Science and Technology, Hunan Institute of Humanities, Science and Technology, Loudi, China
Email: gjguo@163.com

Fei Yu[1] and Zhigang Chen[2] and Dong Xie[2, 3]
1. Guangdong Province Key Lab of Electronic Commerce Market Application Technology
Guangdong University of Business Studies, Guangzhou, China
2. College of Information Science and Engineering, Central South University, Changsha, China
3. Department of Computer Science and Technology, Hunan Institute of Humanities, Science and Technology, Loudi, China
Email: hunanyufei@126.com, czg@csu.edu.cn, lgxzy@163.com

***Abstract*—In order to select the proper one from the web service candidate sets which are provided by semantic matching, a method for web service selection with QoS (quality of service) constraints is proposed. In this paper, firstly, the Four-Level Matching Model for Semantic Web Service based on QoS Ontology, which includes Application Domain Matching, Description Matching of Service, Function Matching of Service and QoS Matching, is presented, then, the QoS ontology for web service is constructed and the QoS description information with the QoS concept in OWL-S (Web Ontology Language for Services) model is extended, thirdly, the extensible OWL-SQ model for service description with service semantic and QoS description is presented, and it is QoS supporting and constraining. Then based on decision theory and normalized algorithm, the dimensionless method of QoS properties for un-numerical value type and numerical value type in OWL-SQ model is proposed. Finally, the algorithm for constructing matching matrix of QoS and the service selection method are presented, to evaluate and select optimization candidate web service comprehensively. Test result shows that the efficiency and accuracy of service selection are improved with the service selection method based on OWL-SQ model.**

***Index Terms*—QoS ontology, semantic web services, service selection, Matching Model, Web Ontology Language for Services**

I. INTRODUCTION

Web service is an application component distributed, self-described and modularized in Web environment. With the rapid growth of Web service quantity, more and more stable easy-to-use Web services are shared on the network. Facing the huge number of services, how to discover services, which are satisfied with needs for developing Web service, has become a bottleneck problem. Semantic Web Service (SWS) is a good semantics and a very important field based on semantic Web and ontology. Its function is to implement extensible automatic or semi-automatic service discovery and composition in the dynamic, open and heterogeneous network environment with the support of abundant semantics.

The dynamic e-commerce requires seamless and integrate business process, application and Web services. Due to the dynamic and unpredictable internet, it is a crucial and significant challenge to provide QoS on the internet. Some aspects produce internet QoS standard requirement such as communication mode changes, denial of service attack, infrastructure failure of performance and low Web agreement on the Web and safety problems. QoS problem will cause important affairs application to suffer unacceptable performance [1]. Providing guaranteed QoS is key for the commercial application of success [2].

In some Web-services-related standards are proposed such as UDDI, SOAP, XML, WSDL, JWSFL, XLANG, BPEL4WS, DAML-S and OWL-S, but only DAML-S and OWL-S mention service quality, which includes quality attributes such as the degree of quality and quality guarantee et al. In the non-functional attributes, the service quality attributes provide a feasible reference for the quantitative evaluation for publishers. Some studies also provide relevant Web service quality evaluation, emphasizing the importance of the QoS Web service. Reference [3] provides service quality evaluation model, which includes cost, time, reliability and fidelity. Reference [4] describes a Web service specification that should include syntax, semantics and QoS; and defines time, cost and reliability as attributes of Web services quality.

In order to describe functions of Web service and the semantic information of the parameters, people gradually introduce ontology, which defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class members). Ontology is a formal and explicit specification of a shared conceptualization of a domain of interest. By

This work is supported by the Hunan Provincial Natural Science Foundation of China under Grant No.07JJ6113, Hunan Provincial 15th Educational Science Plan of China under Grant No.XJK08AXJ002, Hunan Provincial Important Constructing Subject of Computer Application Technology of China.

doi:10.4304/jcp.6.2.377-386

defining concepts and relations among concepts it describes semantic information [5]. Ontology may be explained with a directed tree diagram. In which nodes describe concepts, and the connections between the nodes describe the corresponding semantic relation among concepts. As an effective model, to express structure and semantic model of concept levels, ontology is very suitable for semantic Web knowledge representation. In ontology, different marks of the same meaning express the same concept. Therefore, in the semantic Web and Web service matching, ontology has a very important position, and the semantic Web is the foundation of the information sharing and exchanging in semantic levels.

In order to select the proper one from the web service candidate set provided by semantic matching, a method for web service selection with QoS constraints is proposed. Firstly, the *Four-Level Matching Model for Semantic Web Service based on QoS Ontology* is presented, and the QoS ontology for web service is designed. Then, the method extends the QoS description information with the QoS concept in OWL-S (Web Ontology Language for Services) model, and the extensible OWL-SQ model for service description with service semantic and QoS description is presented, and it is QoS supporting and constraining. Then based on decision theory and normalized algorithm, the dimensionless method for QoS properties of un-numerical value type and numerical value type in OWL-SQ model is proposed. Finally, the algorithm for constructing matching matrix of QoS and the service selection method are presented, to evaluate and select optimization candidate web service comprehensively.

II. FOUR LEVEL MATCHING MODEL FOR SEMANTIC WEB SERVICE BASED ON QOS ONTOLOGY

In order to realize service selection for supporting QoS constraints, this paper presents the *Four-Level Matching Model for Semantic Web Service based on QoS Ontology*(FLMM), which is showed in figure 1. The *Matching Engine* works with the concepts of domain ontology and QoS ontology; it executes several matching operations in sequence, which includes *Application Domain Matching, Description Matching of Service, Function Matching of Service, QoS Matching,* for the description of the user's service request and registered service advertising. Its basic matching algorithm is listed following.

Step 1: *Application Domain Matching*. It is based on domain ontology. Firstly, computes the application domain similarity degrees between service request and register service advertising. Then uses the given thresholds of domain similarity degrees to filter the irrelevant services for user service requests, and obtain the candidate set C_{adm} by big granularity level.

Step 2: *Description Matching of Service.* It employs TFIDF[6] algorithm to compute the similarity degrees between vector of service request description and vectors of service advertising description in the candidate service set C_{adm}. Finally, according to the given thresholds, it executes the filter in the second stage to obtain candidate service set C_{dms}.

Step 3: *Function Matching of Service.* It employs the improved Kuhn-Munkres(KM)[7] algorithm to compute the similarity degrees of IOPE functional parameters in service request description and service advertising description for the service candidate set C_{dms} based on bipartite graph. According to given thresholds, it executes the filter in the third stage to obtain candidate service set C_{fms}.

Step 4: *QoS Matching.* It executes the dimensionless of various non-numerical and numerical quantity attributes for candidate service set based on QoS ontology and domain ontology. The algorithm and service selection method are constructed by QoS matching matrixes. According to given thresholds, it executes the filter in the forth stage to obtain candidate service set C_{QoS}.

Step 5: *Sorting of Service.* According to a sorting rule, it executes the candidate service set C_{QoS} to return the candidate service set that it satisfied with the given thresholds for service requesters.

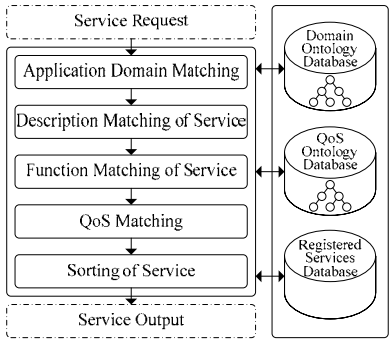

Figure 1 four-level matching model for semantic Web Service based on QoS ontology

A. Application Domain Matching

Application Domain Matching is based on domain ontology and the service categorical standards such as NAICS (North American Industry Classification System) [8] and UNSPSC (Universal Standard Products and Services Classification) [9] etc. Firstly, it computes the application domain similarity degrees between service request and register service advertising. Then, uses the given thresholds of domain similarity degrees to filter the irrelevant services for user service requests by big granularity level, and obtain the candidate set C_{adm}. Its matching rules are the following: If $Sim(WS_r, WS_i) \geq \partial_{adm}$, then WS_i and WS_r is satisfied with the application domain matching, and WS_i is joined candidate service set C_{adm}. Where WS_r is service request, WS_i is discretional service in the registered services library, ∂_{adm} is the threshold of the domain similarity degree.

B. Description Matching of Service

Description Matching of Service is based on domain ontology, firstly, the character segmentation processing and the feature term extraction for service description documents is executed, and the vectors of service

description document are constructed, and the *Vector Space Model(VSM) of Service Description* is shown in figure 2. Then, it employs TFIDF algorithm to compute Cosine(θ) between the service request description vector W_{D_r} and service advertising description vector W_{D_a} as the similarity degrees of the two service description documents.

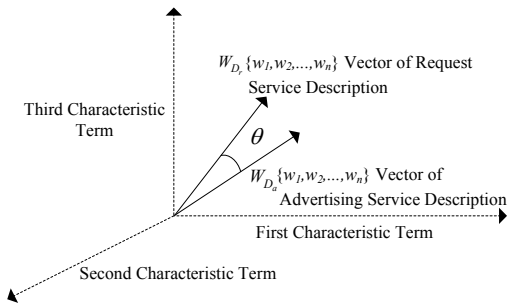

Figure 2 Vector Space Model of Document of Service Description and Similitude Degree of $Sim(W_{D_a}, W_{D_r})$

Definition 1: The service description document D={t_1, t_2,..., t_i, ...,t_n},where t_i denotes the feature terms of service description, n denotes the number of feature terms.

Definition 2: The service description document vector, namely, is the weight vector W_d = {w_1,w_2,... ,w_j ,...,w_n}, where w_j denotes the weight of t_i in service description document D, n denotes the number of the terms in the service description documents.

Definition 3: Hypothesize the frequency of term t as *TF* in the service description documents. Well then the weight w_{ij} and *IDF*(Inverse Document Frequency) employee formula (1) and formula (2) to compute respectively:

$$w_{ij} = tf_{ij} \times idf_j \quad (1)$$

$$idf_j = \log_2^{N/df_j} \quad (2)$$

where, w_{ij} denotes the weight of term j in the service description i; tf_{ij} denotes the number of term j in the service description i; idf_j is the total number of documents divided by the number of documents with term j; N denotes the total number of documents, df_j denotes the number of documents with term j, log_2 limits idf_j .

Set the weight vector of the service request description document as $\overline{r}$ ={$W_{r,1}$, $W_{r,2}$,…$W_{r,n}$}, the weight vector of the service advertising description document as $\overline{a_j}$ ={$W_{j,1}$, $W_{j,2}$,…$W_{j,n}$}, θ is the Angle between $\overline{r}$ and $\overline{a_j}$, then, the similarity degrees between $\overline{r}$ and $\overline{a_j}$ can be expressed as Cosine(θ), which is as formula(3), the smaller angle, the bigger similarity degrees.

$$Sim(\overline{r},\overline{a}) = \frac{\overline{r}\cdot\overline{a}}{|\overline{r}|\times|\overline{a}|} = \frac{\sum_{i=1}^{n} W_{r,i}\times W_{a,i}}{\sqrt{\sum_{i=1}^{n}(W_{r,i})^2\times(W_{a,i})^2}} \quad (3)$$

where, $w_{r,i}$ is the weight of term i in the document D_r, $w_{a,i}$ is the weight of term i in the document D_{A_i} . If $Sim(\overline{r},\overline{a}) > \partial_{dms}$, then D_r is similar with D_{A_i} , else dissimilarity, where ∂_{dms} is a similarity degrees threshold.

C. Function Matching of Service

Function Matching of Service is based on domain ontology to match the function requests of service requesters with IOPE of service providers, it searches the satisfied services according to the similarity degrees. We transfer the service function matching to the best matching of the weighted bipartite graph[10], and use KM algorithm to compute the similarity degrees of function parameters in service interface.

Definition 4: The maximum matching. Set bipartite graphs as a undirected graph G=(V_1, V_2 ,E), where, V_1 and V_2 is a partition in graph G nodes set, and $V_1 \cap V_2=\varphi$, E is edges set; a matching M of the bipartite graph G is a non-null subgraph of G, any edges of M are not adjacent in G. If any matching M' of the bipartite graph, $|M'|\leq|M|$, then, M is the maximum matching of G.

Definition 5: The best matching. Suppose there is a weighted bipartite graph K=(G, w), G is a bipartite graph, and $w(x_i, y_j)$ is the weight of edge (x_i ,y_j) ($x_i \in V_1$, $y_j \in V_2$). If M is the maximum matching of G and the sum of weight of M is the biggest, then M is the best matching of K.

Definition 6: The function matching. Suppose O_r (r_1, r_2 , ... , r_n) is made up of n output parameters, and it means the concept set of output parameters of service request WS_r, O_a(a_1 ,a_2 , ... ,a_m) is made up of m output parameters, and it means the concept set of output parameters of service advertising WS_a . Then function matching can be changed into the similarity degrees $SF_{i,j}(a_i,r_j)$ between service parameters of WS_r and WS_a, where, a_i *is* one of the output parameters set O_a(a_1 ,a_2 , ... ,a_m) of service advertising WS_a , and r_j is one of the output parameters set O_r (r_1, r_2 , ... , r_n) of service request WS_a.

To compute the biggest similarity degrees of two concept sets is the same as the best matching of the weighted bipartite graph. This paper uses improved KM algorithm to solve the best matching problem of service bipartite graph. KM algorithm inputs service bipartite graph $G_{m,n}$ =(O_r, O_a, E), and outputs service parameter best match M [10]. We use KM algorithm to compute the similarity degrees of two parameter sets, and according to the given thresholds obtaining candidate service set C_{fms}.

D. QoS Matching

We establish QoS ontology of Web service, scalable OWL-SQ service description model and QoS matching matrix to execute dimensionless processing to compute the normalized value of QoS attributes of various service in candidate service set C_{fms} by normalized algorithm for various non-numeric and numeric quantity attributes. According to the given the threshold ∂_{QoS} ,the candidate

set C_{QoS} should be returned, whose normalized value is more than the threshold.

E. Sorting of Service

Do sorting to C_{QoS} based on service quality or service function, and send the sorting to service requester. Apply Borda[11] sorting algorithm can get a fair sorting result.

III. QOS ONTOLOGY

The QoS ontology should be divided into top-level ontology, middle-level general ontology and bottom-level application ontology. The top-level ontology describes the most basic quality concepts, and defines the basic object attributes and data attributes of QoS. The middle-level general ontology defines the QoS concepts of universal field. The bottom-level application ontology describes the QoS concepts of specific domain.

A. QoS top-level ontology

The top-level ontology describes the contents with a QoS parameter, the QoS top-level ontology is shown in Figure 3. Including the main concepts (classes) and attributes such as *QoSProfile*, *QoSAttribute*, *QoSMetric*, *QoSValue*, *QoSUnit*, *QoSTendency, QoSRelationship*, *QoSAggregated* and *QoSWeight* et al.

- *QoSProfile*: it is the top QoS model, which employs the concept of *hasQoSAtrribute* attribute to define all the QoS parameters of a service.
- *QoSAttribute*: it has two direct subclasses: *CommonQoSAttribute* and *DomainQoSAttribute*, which respectively connect *QoSAttribute* through the two properties (*hasCommonQoSAtrribute* and *hasDomainQoSAtrribute*).
- *QoSMetric*: it defines each measure of QoS attributes, which should be divided into numeric and non-numeric. Numeric type includes exactitude and interval types, the non-numeric type is divided into text, level and Boolean type. *QoSMetric* of a QoS attribute may be characters, numbers, Boolean or string.
- *QoSValue*: it indicates the QoS attribute measure value, numeric and non-numeric value are indicated by integer and real.
- *QoSUnit*: it describes the units of properties, multiple attributes involve multiple units, according to the requirement they may be expanded.
- *QoSTendency*: it indicates the influence which is the change of QoS attribute value for service quality. It connects *QoSAttribute* by *hasTendency* attributes. *HasTendency* describes the trend of QoS attributes. It has two values increase and decrease. If *hasTendency* is increased, this attribute is positive for the overall QoS; if *hasTendency* is decreased, the parameters are negative effect on overall QoS.
- *QoSRelationship*: it indicates the relationship between a QoS parameter and other parameters.
- *QoSAggregated*: it connects QoS attributes by way of *isAggregated* attribute, which indicates some QoS attribute is composed of two or more other parameters. For example, the response time includes delays and request processing time.
- *QoSWeight*: it indicates the weight of QoS attribute in the *QosProfile*.

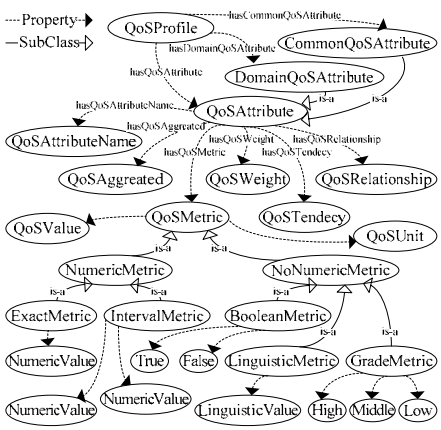

Figure 3 QoS top ontology

B. QoS middle-level general domain ontology

Considering QoS definition in reference [12], the author constructs the QoS middle-level general domain ontology with Protege3.4 shown in Figure 4. The main parameters of the QoS include: Accessibility, Accuracy, Availability, Capacity, Economic, ExceptionHanding, Integrity, Interoperability, Network_related, Performance, Reliability, Reputation, Robustness, Scalability, Security, Stability.

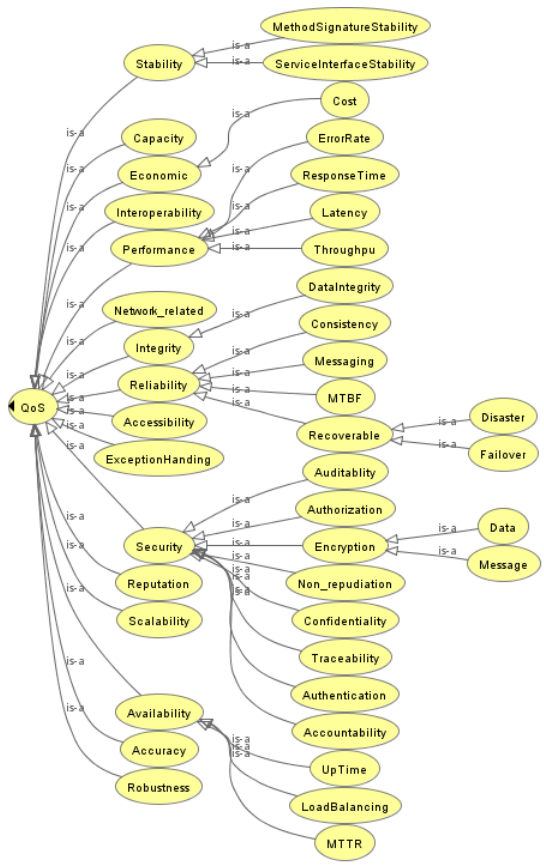

Figure 4 QoS middle-level generic domain ontology

IV. THE OWL-SQ WITH QOS CONSTRAINT EXTENTDED MODEL

A. OWL-S description model

OWL-S [13](Web Ontology Language for Services)is a kind of OWL ontology for describing Web service attributes and function in the semantic Web. OWL-S includes a set of ontology: On the top-level is a service type which includes three attributes such as *presents*, *describedBy* and *supports*; on the second level are the three ontologies: *ServiceProfile*, *ServiceModel*, and *ServiceGrouding*. Corresponding to the three attributes: *presents*, *describedBy*, and *supports* of Service. *ServiceProfile* describes the services what to do, it gives a set of information for service match; *ServiceModel* describes the service how to do, which is to execute logical sequence of service; *ServiceGrounding* describes how to access service, i.e. the access service communication protocol and other details. A Service is described by a *ServiceModel* at most; a *ServiceGrounding* must be associated with a service.

ServiceProfile describes three basic aspects of the service: a) related information of service providers, mainly including contact information of service providers; b) The function information of service *IOPE* (Input, Output, Precondition, Effect), which is described by state transform function, in which *IO* expresses the input and output which service needs, *PE* expresses the pre-conditions and results which service needs. c) Other characteristics of the service includes the information which is relevant services, for example, the quality of service, service information, expanding or custom service parameters, maximum response time, service and geographical scope, etc. *ServiceProfile* is bidirectional, and service providers can express the service function by the *Profile* ontology. Service requesters or service query agents can describe service inquires by *Profile*, which judges whether a service corresponds to user's requirements. *ServiceProfile* is unconcerned the registered model. It supports the UDDI and P2P various registered models.

B. OWL-SQ extended model with QoS constraints

1) Service semantics

OWL-SQ model includes service semantics and the quality description model [14], mainly defines the basic information, input/output parameters, precondition/ postposition conditions parameters, QoS and the related semantic information for the web service.

Definition 7: The semantic Web services (SWS) with QoS constraints may be described as five tuples:

$SWS_{OWL\text{-}SQ} = \{N, D, IOPE, QoS, O\}$

Where, N is the service name or service identifier; D is the service description; *IOPE* is input/output parameters of the service and the service for achieving previous conditions and result sets, $IOPE = \{ I, O, P, E\}$, where, $I = \{in_1, in_2, ..., in_n\}$, $in_i=\{type, value\}$, $O = \{out_1, out_2, ..., out_n\}$, $out_i =\{type, value\}$, $P = \{pre_1, pre_2, ..., pre_n\}$, $E = \{eff_1, eff_2, ..., eff_n\}$; *QoS* is the service quality evaluation model; O is the service domain ontology for supporting *SWS*, *N*, *D* and *IOPE*, and their concepts and their relationships and rules are in O.

2) QoS description model

This is a simple but suitable QoS model, which is automatic to compute QoS measure. The model includes the default QoS attributes: Accessibility, Availability, Cost, ResponseTime, ErrorRate, Throughput, Reliability, Reputation, Robustness, Security.

Definition 8: QoS description model

It is an extendable vector, which is used to define QoS parameters of service advertising and service request. It is expressed as : $QoS = \{qos_1, qos_2, qos_3, ..., qos_i\}$, $i \in N$; where, qos_i indicates ith quality attribute, QoS_I indicates quality attributes sets, QoS_D indicates each service default quality attributes set, QoS_U indicates user-defined quality attributes, so a service request quality attributes $QoS_R=QoS_D \cup QoS_U$. This default quality attribute sets QoS_D = {*Accessibility, Availability, Cost, ResponseTime, ErrorRate, Throughput, Reliability, Reputation, Robustness, Security*}, Where:

- *Accessibility*: It shows the extent which can provide service for the Web service and expresses the success rate of instantiated service at a point of time. A highly scalable system can be built to obtain high accessibility of Web service.
- *Availability*: It means whether the Web service exists or it is ready for immediate use, and shows the possibility of service available. A larger probability value indicates that service is always available; a smaller value indicates that it can not predict whether the services are available at a particular time.
- *Cost:* It shows the cost for which requesters obtain services, that is to ensure that service requesters can achieve tasks within the budget range.
- *ResponseTime*: It is a general performance measure. *ResponseTime* includes two parts: the delay time and the processing time. For Web services, its processing time is the time that Web service accepts the request message to response message return. Its delay time includes the creation, transmission, and processing time of request information.
- *ErrorRate*: The error generated in transmission system is usually measured with the error rate of bit, error rate of packet, loss rate of packet, etc. Multimedia application data such as voice and moving image can tolerate the occurrence of errors, but the data transfer demands highly on error rate such as bank transfers, stock market, scientific data and control instructions, which do not allow any mistakes.
- *Throughput*: In the case of no frame loss, the *Throughput* is the maximum rate that the servers and firewalls and other devices can accept. Due to a variety of low efficiency factors, throughput is generally lower than the link bandwidth.
- *Reliability*: It is the possibility whether a user can obtain service when the user asks for service and it expresses the extent of the services maintenance and the service quality. Failure frequency in a month or year is a measure of the reliability of Web

services standards. In another sense, *Reliability* is a situation where the service requestors and service providers send and receive messages that can be guaranteed and ordered to delivery. An important indicator of the *Reliability* is the possibility which users obtain services when they request the service. Its most common assessment approach is the number of days which the service normally operates in a year..

· *Reputation*: It is the service evaluation (confidence) which is received through the monitoring of a neutral body or the service evaluation (satisfaction) of a service requestor. Credibility can reduce the impact of services false QoS information to services choices.

· *Robustness*: It is the robustness of system. It is the key of survival of the system in exceptional circumstances, which can be divided into stability robustness and performance robustness. For example, the service system can stably operate without crashes in some circumstances such as input error, network overload, malicious attacks, that is, the robustness of the service system

·*Security*: It is an aspect of the quality of Web service, it validates the involved message encryption and access control to provide confidentiality. According to the difference of service requestors, service providers can use different methods to provide security, which can have different levels.

Definition 9: The QoS description model of candidate service

Suppose there is n selected candidate services by functional semantic matching, the candidate service set is expressed as: $QoS_C = \{Q_{C1}, Q_{C2}, \ldots, Q_{Cn}\}$, where $Q_{Ci} = ((q_{i1}, v_{i1}), (q_{i2}, v_{i2}), \ldots, (q_{im}, v_{im}))$ is the quality of service of the ith-candidate Web services, $1 \le i \le n$, $q_{ij}(1 \le j \le m)$is the property name of the first jth QoS quality in Q_{Ci}, v_{ij} shows the corresponding quality attribute values which is provided by services ads promise.

Definition 10: The QoS description model of service request

$QoS_R = ((q_1, w_1, v_1), (q_2, w_2, v_2), \ldots, (q_n, w_n, v_n))$

Where, $q_k(1 \le k \le n)$is the kth quality attribute name in service requests, w_k indicated the weight which requester assigns to the quality attributes, v_k is requester's expectations of the quality attribute.

V. THE SELECTION METHOD OF SERVICES WITH QOS CONSTRAINTS

A. QoS attributes dimensionless method

Since the metric way and the value of measure of different types of QoS attributes are different, in order to carry out numeric matching; firstly, the QoS attributes need be quantified. Meanwhile, since a variety of properties of dimension are different, they can not be compared and calculated with each other, and they need to carry out unified treatment [15]. That is dimensionless method.

According to the different types of quality attributes values, they can be divided into numerical, text, grade and Boolean type, etc. *a) Numeric type:* numeric type is divided into exactitude type and interval type, precision-type determines the QoS property values with a certain value, interval-type describes a property's value with the interval bounds. *b) Descriptive text type*: It uses the defined orderly and limited language words to measure the QoS attribute values. For example, reputation can be expressed as {very high, high, medium, low, very low}. *c) Class type*: for example, the service level can be expressed as {1, 2, 3, 4, 5, 6}. *d) Boolean type*: it is expressed by using {Yes, No} or {True, False}.

1) Quantify of non-numeric attribute values

For the descriptive text type such as the credibility of {high, high, medium, low, low}, it is expressed by using {5, 4, 3, 2, 1} respectively; for the Boolean type (Yes, No), it is expressed by using (1, 0) respectively.

2) The dimensionless method of the numerical value

Suppose candidate service sets by functional match $S = (S_1, S_2, S_3, \ldots, S_n)$, S_i has m QoS attributes, $1 \le i \le n$, then get an $n \times m$ order decision matrix Q , where each row represents a candidate service correspond to each QoS property value, each column represents attribute values of all candidate services.

$$Q = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1m} \\ q_{21} & q_{22} & \cdots & q_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ q_{n1} & q_{n2} & \cdots & q_{nm} \end{bmatrix}$$

According to the tendency of service quality attributes, it can be divided into two kinds: increase quality attributes and decrease quality attributes. The increase quality attribute means: the larger the QoS values is, the higher the quality is, such as credibility, availability and reliability. Decrease quality attributes means that the greater QoS values is, the lower the quality is, such as the execution time and costs.

a) Normalization of the exactitude type

If the quality attribute value is exactitude, the increase attributes and decrease attributes are normalized by the formula (4) and (5) respectively.

$$v_{i,j} = \begin{cases} \dfrac{q_{i,j} - q_j^{\min}}{q_j^{\max} - q_j^{\min}} & \text{if } q_j^{\max} - q_j^{\min} \neq 0 \\ 1 & \text{if } q_j^{\max} - q_j^{\min} = 0 \end{cases} \quad (4)$$

$$v_{i,j} = \begin{cases} \dfrac{q_j^{\min} - q_{i,j}}{q_j^{\max} - q_j^{\min}} & \text{if } q_j^{\max} - q_j^{\min} \neq 0 \\ 1 & \text{if } q_j^{\max} - q_j^{\min} = 0 \end{cases} \quad (5)$$

Where $v_{i,j}$ is the normalized results of jth attributes of the ith service; $v_{i,j}$ is the maximum value of the jth column of the quality matrix Q, $q_j^{\max} = \max(q_{i,j})$, $1 \le i \le n$; $q_j^{\max}$ is minimum value of the j-column of the quality matrix Q, $q_j^{\min} = \min(q_{i,j})$, $1 \le i \le n$. For the formula (4) , if $q_j^{\max} - q_j^{\min} \neq 0$ is an increasing function, $q_{i,j}$ is $q_j^{\min}$, the evaluation value is 0, $q_{i,j}$ is the closer $q_j^{\max}$,

the higher the value is, when $q_{i,j} = q_j^{\max}$, the evaluation value is 1; When $q_j^{\max}$ - $q_j^{\min}$ = 0, this shows that property values are equal, let the evaluation value be 1.

Example 1: Let the candidate sets of Web services which complete task *t* is a total of 10, its execution time attribute value (in milliseconds) in the jth columns of the quality matrix *Q*, which indicate respectively q_j = (180, 160, 90, 150, 100, 120, 220, 220, 260, 280), since the execution time is the type of reduction quality attribute, we use formula (5) to normalize, where $q_j^{\max}$ = 280, $q_j^{\min}$ = 90, after treatment of normalization, the attribute value of execution time is v_j = (0.53, 0.63, 1.00, 0.68, 0.95, 0.84, 0.32, 0.32, 0.11, 0.00).

b) Normalization of the interval type

For the interval QoS quality attributes, the approach employs similar exactitude type to normalize the upper and lower limits of the range respectively, and choose finally the average value as a synthetical evaluation. Since the infinite of the upper and lower limits is unreasonable, it is not considered. For the jth QoS attribute of the ith service, if the interval is used to describe $q_{i,j} = [\underline{q_{i,j}}, \overline{q_{i,j}}]$, and $\underline{q_{i,j}}$ and $\overline{q_{i,j}}$ are lower boundary and upper boundary. For positive quality attributes use formula (6) (7) (8) to normalize; for the reverse quality attributes use formula (9) (10) (11) to normalize.

$$\underline{v_{i,j}} = \begin{cases} \dfrac{\underline{q_{i,j}} - \underline{q_j^{\min}}}{\underline{q_j^{\max}} - \underline{q_j^{\min}}} & \text{if } \underline{q_j^{\max}} - \underline{q_j^{\min}} \neq 0 \\ 1 & \text{if } \underline{q_j^{\max}} - \underline{q_j^{\min}} = 0 \end{cases} \quad (6)$$

$$\overline{v_{i,j}} = \begin{cases} \dfrac{\overline{q_{i,j}} - \overline{q_j^{\min}}}{\overline{q_j^{\max}} - \overline{q_j^{\min}}} & \text{if } \overline{q_j^{\max}} - \overline{q_j^{\min}} \neq 0 \\ 1 & \text{if } \overline{q_j^{\max}} - \overline{q_j^{\min}} = 0 \end{cases} \quad (7)$$

$$v_{i,j} = \frac{\underline{v_{i,j}} + \overline{v_{i,j}}}{2} \quad (8)$$

Eq. (6) is a formula where positive quality attributes normalize the lower bound $q_{i,j}$, where $\underline{v_{i,j}}$ is the normalized value, $\underline{q_j^{\max}}$ is the maximum of all the lower bound of the attribute, $\underline{q_j^{\min}}$ is the minimum of all the lower bound of the attribute. Eq. (7) is a formula where positive quality attributes normalize upper bound $q_{i,j}$, where $\overline{v_{i,j}}$ is the normalized value, $\overline{q_j^{\max}}$ is the maximum of all the upper bound of the attribute, $\overline{q_j^{\min}}$ is the minimum value of all upper bound of the attribute. Eq. (8) is a formula where positive quality attributes finally assess $v_{i,j}$, the value is the mean results of formula (6) (7).

$$\underline{v_{i,j}} = \begin{cases} \dfrac{\underline{q_j^{\max}} - \underline{q_{i,j}}}{\underline{q_j^{\max}} - \underline{q_j^{\min}}} & \text{if } \underline{q_j^{\max}} - \underline{q_j^{\min}} \neq 0 \\ 1 & \text{if } \underline{q_j^{\max}} - \underline{q_j^{\min}} = 0 \end{cases} \quad (9)$$

$$\overline{v_{i,j}} = \begin{cases} \dfrac{\overline{q_j^{\max}} - \overline{q_{i,j}}}{\overline{q_j^{\max}} - \overline{q_j^{\min}}} & \text{if } \overline{q_j^{\max}} - \overline{q_j^{\min}} \neq 0 \\ 1 & \text{if } \overline{q_j^{\max}} - \overline{q_j^{\min}} = 0 \end{cases} \quad (10)$$

$$v_{i,j} = \frac{\underline{v_{i,j}} + \overline{v_{i,j}}}{2} \quad (11)$$

Eq. (9) is a formula where positive quality attributes normalize the lower bound $q_{i,j}$, where $\underline{v_{i,j}}$ is the normalized value, $\underline{q_j^{\max}}$ is the maximum of all the lower bound of the attribute, $\underline{q_j^{\min}}$ is the minimum of all the lower bound of the attribute. Eq. (10) is a formula where reverse quality attributes normalize upper bound $q_{i,j}$, where $\overline{v_{i,j}}$ is the normalized value, $\overline{q_j^{\max}}$ is the maximum of all the upper bound of the attribute, $\overline{q_j^{\min}}$ is the minimum value of all upper bound of the attribute. Eq. (11) is a formula where reverse quality attributes finally assess $v_{i,j}$, the value is the mean results of formula (9) (10).

If the services of the candidate services sets describe the same QoS attributes, there exists situation which has exactitude value and interval value, the exactitude value can be looked as interval value with the same upper and lower limits, using the formula (6) - (11) to normalize them.

c) Discrete normalization

For discrete data, the normalization algorithm as follows:

$$v_{i,j} = 1 - \frac{I_r}{t-1} \quad (12)$$

Where, *t* is the number of discrete data, I_r is the index position of a certain discrete value, $I_r \in N$ and 0 $\leq I_r \leq$ (*t*-1). If the elements are in descending order, when *r* is in the first index position, I_r = 0, $v_{i,j}$ = 1; when *r* is in the final index position, I_r = *t*-1, $v_{i,j}$ = 0; for all *r*, the $v_{i,j}$ are between 0 and 1.

B. Matching matrix construction algorithm of the QoS attribute

Matching matrix construction algorithm:

Input: counter *x* = 0; service request $QoS_R = ((q_1, w_1, v_1), (q_2, w_2, v_2), ..., (q_m, w_m, v_m))$

Output: return a S_i to the user with the threshold value greater than user set.

Step1: According to the *m* quality attributes of service requests, calculate the semantic matching degree *D* of the candidate services corresponding to the quality attribute. If there exists the semantic matching degree of the default quality parameters *D* = *fail* in QoS_D, this means the

candidate service quality does not meet the user's request, the candidate service should be deleted, so $x = x + 1$;

Step2: According to the order of quality parameters in QoS_R, we rearrange quality parameters in Q_{Ci} and normalize the quality attributes in Q_{Ci} according to the above-mentioned dimensionless algorithm, and form a dimensionless matching matrix $Q' = \{Q_{C1}, Q_{C2}, ..., Q_{C(nx)}\}$.

$$Q' = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1m} \\ v_{21} & v_{22} & \cdots & v_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ v_{(n-x)1} & v_{(n-x)2} & \cdots & v_{(n-x)m} \end{bmatrix}$$

Q' is an $(n - x) \times m$ matrix, each row is a dimensionless value of the various quality attributes in a service, and each column corresponds to a same quality attribute of different services.

Step3: multiply the weight value by which the user assigns various quality attributes and the dimensionless matching matrix, according to the formula (13) calculate the matching degree for each service Q_{Ci} and the values of request Q_R:

$$QoS(s_i) = \sum_{j=1}^{m} (v_{ij} \times w_j) \qquad (13)$$

Where, $QoS(s_i)$ indicates the final integrated QoS value of s_i; w_j shows the weight of the jth quality attribute, $w_j \in [0,1]$, $\sum_{j=1}^{m} w_j = 1$.

Step4: return the number of service users require according to descending $QoS(s_i)$ values or the threshold users set, return a S_i greater than the threshold value to the user. Where, the formula (14) for calculating the optimal quality of Web services is:

$$S_{\text{optimization}}(t) = MAX_{i=1}^{n}(QoS(s_i)) \qquad (14)$$

VI. SIMULATION EXPERIMENTAL VERIFICATION

In order to verify the feasibility and the superiority of the semantic Web service selection algorithm with QoS constraints, we use the following examples to test.

In tests, we use instance data, select the default service quality attributes QoS_D such as the response time, service costs, availability, accessibility, reliability, security and reputation to constitute the QoS parameters where response time and services cost in milliseconds and Yuan, availability, accessibility and reliability are expressed by a percentage, the security level is an integer range as [1, 10], and credibility rating value is an integer range as [1, 5]. QoS constraints of services requester show in Table I, according to different constraint degrees of quality attributes of different service requester, they correspond to different weight values.

TABLE I. QoS CONSTRAINTS OF SERVICE REQUEST

	response time		***service costs***		***availability***		***accessibility***	
	expectation	***weight***	***expectation***	***weight***	***expectation***	***weight***	***expectation***	***weight***
R1	48	0.2	68	0.2	0.85	0.1	0.78	0.1
R2	40	0.1	75	0.1	0.9	0.2	0.9	0.2

Continue of TABLE I

	reliability		***security***		***reputation***	
	expectation	***weight***	***expectation***	***weight***	***expectation***	***weight***
R1	0.88	0.1	8	0.1	4	0.2
R2	0.9	0.2	8	0.1	4	0.2

Again let four Web services of candidate Web service set by the function semantic matching as shown in Table II.

TABLE II. THE CANDIDATE WEB SERVICES

	response time	***service costs***	***availa-bility***	***accessi-bility***	***relia-bility***	***secu-rity***	***repu-tation***
S1	50	70	0.9	0.8	0.9	9	5
S2	40	80	0.95	0.9	0.9	9	4
S3	70	60	0.85	0.8	0.8	7	3
S4	90	50	0.75	0.7	0.7	9	5

Take R1 for example, we get:

$$Q = \begin{bmatrix} 50 & 70 & 0.90 & 0.8 & 0.9 & 9 & 5 \\ 40 & 80 & 0.95 & 0.9 & 0.9 & 9 & 4 \\ 70 & 60 & 0.85 & 0.8 & 0.8 & 7 & 3 \\ 90 & 50 & 0.75 & 0.7 & 0.7 & 9 & 5 \end{bmatrix}$$

The corresponding quality attributes are response time, service costs, availability, accessibility, reliability, security, and credibility. According to each column of attributes, if we put the first and second column reduction properties into the formula (5), the third, fourth, fifth, sixth, seventh column incremental properties into the formula (4), which calculates and gets the final the normalization weight matrix:

$$Q' = \begin{bmatrix} 0.16 & 0.07 & 0.08 & 0.05 & 0.10 & 0.10 & 0.20 \\ 0.20 & 0.00 & 0.10 & 0.10 & 0.10 & 0.10 & 0.10 \\ 0.08 & 0.13 & 0.05 & 0.05 & 0.05 & 0.00 & 0.00 \\ 0.00 & 0.20 & 0.00 & 0.00 & 0.00 & 0.10 & 0.20 \end{bmatrix}$$

Put the weight W = (0.2, 0.2, 0.1, 0.1, 0.1, 0.1, 0.2) into the formula (13), get the value of comprehensive quality of each service:

$$QoS = \begin{bmatrix} 0.75 \\ 0.70 \\ 0.36 \\ 0.50 \end{bmatrix}.$$

The final matching results for service requests R1, R2 are shown in Table 3.

TABLE III. MATCHING RESULT

	S1	***S2***	***S3***	***S4***	***The optimal matching service***
R1	0.75	0.70	0.36	0.50	S1
R2	0.86	0.90	0.41	0.40	S2

This example shows that the algorithm can recommend different services for different requestors according to different service requests. It can conduct further value matching based on the QoS semantic parameters matching , if the exact matching can not meet the users' needs, it can select value matching degrees to meet Web services user require from insert and contain matches to return to the user.

VII. RELATED WORK

In [12] a QoS ontology and its vocabulary using the Web Services Modeling Ontology (WSMO) for annotating service descriptions with QoS data are specified, a QoS-based selection of services is proposed by defining quality attributes and their respective measurements along with a QoS selection model.

The work [16] and [17] emphasized the definition of QoS aspects and metrics. In [16], all of the possible quality requirements were enumerated and organized into several categories, including runtime-related, transaction support related, configuration management and cost-related QoS, and security-related QoS. But, they failed to present quantifiable measurements.

In [18] and [19] focused on the creation of QoS ontology models, which proposed QoS ontology frameworks aiming to formally describe arbitrary QoS parameters. Additionally, the work [20], [21], and [22] tries to attempt to conduct a proper evaluation and proposes QoS-based service selection.

In [2] a model for web service discovery with QoS constraints called WSDM-Q is proposed. A set of QoS categorization tModels and a kind of reputation categorization tModel are defined in the model to describe the QoS attributes of a service and the degree of guaranteed QoS delivered by a service respectively. The concept of quantification is introduced in the model to transform between the QoS attributes and QoS categorizations.

In [23] a QoS-Guaranteed and distributed mechanism of Web service discovery is proposed, which supports Web service discovery with QoS constraints and enhances the QoS of service discovery system. Then, a novel three-dimensional QoS model of Web service is introduced, and a Web service-selecting algorithm is proposed based on the novel model.

IX. CONCLUSION

The discovery technology of Web services based on QoS constraints has always been a hot research of Web services technology. This paper points out the main problems of the current Web service description model. In order to find the most satisfactory service in a number of similar services for the service requesters, and on the base of the analysis of OWL-S description model, we extend the QoS description information of the OWL–S by establishing QoS ontologies, and present a extendable OWL-SQ service description model with service semantic and service quality description. For various non-numerical and numerical-type quality attributes in the OWL-SQ model, we give the dimensionless methods. We propose optimal matching matrix should be constructed to evaluate comprehensively the best candidate services. Examples are given to test the method performance. The results show that the service selection method based on OWL-SQ description model service improves the efficiency of the service selection, precision and user satisfaction.

REFRENCES

[1] Mani A, Nagarajan A. Understanding quality of service for Web services. http://www.ibm.com/developerworks/library/ws-quality.html?S_TACT=105AGX52&S_CMP=cn-a-ws

[2] Yang S W, Shi M L. A model for Web service discovery with QoS Constraints. Chinese Journal of Computers, 2005,28(4):589-594.

[3] Cardso J. Adaptive algorithm to predict the QoS of Web processes and workflows. International Conference on Computational Intelligence 2004:490-493

[4] Cafora G, Penta M D, Esposito R, Villani M L. A lightweight approach for QoS-aware service composition. In: Proc of the 2nd International Conference on Service Oriented Computing, New York, USA, 2004, 36-47

[5] Gruber T R. A translation approach to portable ontology specifications. Knowledge Acquisition, 1993, 5(2): 199-220.

[6] Zhou N, Song Z L, Ai W H, Chen L. Research of Three Level Match Method about Semantic WEB Service. Journal of System Simulation, 2009,21(7):2081-2089.

[7] Peng Y X, Ngo C W, Xiao J G. An Approach for Shot Retrieval by Optimal Matching in the Bipartite Graph. ACTA ELECTRONICA SINICA, 2004,32(7):1135-1139.

[8] U.S. Economic Classification Policy Committee (ECPC), Statistics Canada, and Mexico's Instituto Nacional de Estadistica y Geografia. North American Industry Classification System . http://www.census.gov/eos/www/naics/index.html

[9] Electronic Commerce Code Management Association. Universal Standard Products and Services Classification. http://www.unspsc.org/

[10] Deng S G, Yin J W, Li Y, Wu J, Wu Z H. A Method of Semantic Web Service Discovery based on Bipartite Graph Matching. Chinese Journal of Computers, 2008,31(8):1364-1375.

[11] Saari D G. Chaotic elections, a mathematician looks at voting. American Mathematical Society (S0273-0979),2001,45(8):1357-1360.

[12] Wang X, Vitvar T, Kerrigan M, Toma I. A QoS-aware selection model for semantic Web services. In: ICSOC, 2006: 390–401

[13] Martin D, Burstein M, et al. OWL-S: semantic markup for Web services. http://www.ai.sri.com/daml/services/owl-s/1.2/overview/

[14] Li C M, Jiang Y C. Study on semantic Web services discovery with QoS constraint. Computer science, 2007, 34(6):116-121.

[15] Xue C G. The study of QoS-based semantic Web service selection technology. Beijing: Master's degree dissertation of Beijing University of Posts and Telecommunication, 2008.

[16] Ran S. A Model for Web Services Discovery with QoS. SIGecom Exchange, 2003,4(1):1-10.

[17] Lee K C, Jeon J H, Lee W S, Jeong S H and Park S W. QoS for Web Services: Requirements and Possible Approaches. W3C Working Group Note 25, 2003.

[18] Papaioannou I V, Tsesmetzis D T, Roussaki I G and

Anagnostou M E. A QoS Ontology Language for Web-Services. AINA2006.
[19] Tsesmetzis D T, Roussaki I G, Papaioannou I V and Anagnostou M E. QoS Awareness Support in Web-Service Semantics. AICT-ICIW06, 2006, pp.128-128.
[20] Liu Y, Ngu A H H and Zeng L. QoS Computation and Policing in Dynamic Web Service Selection. Proceeding 13th International Conference World Wide Web, 2004.
[21] Mou Y J, Cao J, Zhang S S, Zhang J H. Interactive Web Service Choice-Making Based on Extended QoS Model. Journal of Zhejiang University SCIENCE A, 2006 7(4):483-492.
[22] Menascé D A. QoS Issues in Web Services. IEEE Internet Computing, 2002, 6(6):72-75.
[23] Guo D K, Ren Y, Chen H H, et al. A QoS-Guaranteed and Distributed Model for Web Service Discovery. Journal of Software, 2006,17(11): 2324-2334.

Guangjun Guo was born in 1971. He received his master degree from Wuhan University, Wuhan, China, in 2005. He is currently an associate professor in the Department of Computer Science and Technology, Hunan Institute of Humanities, Science and Technology, Loudi, China. Mr. Guo is a Senior Member of China Computer Federation (CCF). His research interests include semantic Web, services computing and distributed computing.

Zhigang Chen was born in 1964. He received his M.S. degree in computer science from Central South University of Technology, Changsha, China in 1987 and his Ph.D. degree in computer science from Central South University of Technology, Changsha, China in 1998. He is currently a professor, supervisor of Ph.D. and Dean of School of Software of Central South University, Changsha, China. He has published over 100 Journal papers. His research interests include parallel and distributed system, services computing, gird computing and wireless/mobile computing. Mr. Chen is director and Senior Member of China Computer Federation (CCF), member of Pervasive Computing Committee of CCF, member of Guidance Committee of Computer Science and Technology Teaching in Ministry of Education of China.